AD 606844

1

(1)

COMBINATORIAL PROCESSES AND
DYNAMIC PROGRAMMING

Richard Bellman

P-1284

February 24, 1958

Approved for OTS release

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA

CLEARINGHOUSE FOR FEDERAL SCIENTIFIC AND TECHNICAL INFORMATION CFSTI
DOCUMENT MANAGEMENT BRANCH 410.11

LIMITATIONS IN REPRODUCTION QUALITY

ACCESSION # AD 606 844

☒ 1. WE REGRET THAT LEGIBILITY OF THIS DOCUMENT IS IN PART UNSATISFACTORY. REPRODUCTION HAS BEEN MADE FROM BEST AVAILABLE COPY

☐ 2. A PORTION OF THE ORIGINAL DOCUMENT CONTAINS FINE DETAIL WHICH MAY MAKE READING OF PHOTOCOPY DIFFICULT.

☐ 3. THE ORIGINAL DOCUMENT CONTAINS COLOR, BUT DISTRIBUTION COPIES ARE AVAILABLE IN BLACK-AND-WHITE REPRODUCTION ONLY.

☐ 4. THE INITIAL DISTRIBUTION COPIES CONTAIN COLOR WHICH WILL BE SHOWN IN BLACK-AND-WHITE WHEN IT IS NECESSARY TO REPRINT.

☐ 5. LIMITED SUPPLY ON HAND: WHEN EXHAUSTED, DOCUMENT WILL BE AVAILABLE IN MICROFICHE ONLY.

☐ 6. LIMITED SUPPLY ON HAND: WHEN EXHAUSTED DOCUMENT WILL NOT BE AVAILABLE.

☐ 7. DOCUMENT IS AVAILABLE IN MICROFICHE ONLY.

☐ 8. DOCUMENT AVAILABLE ON LOAN FROM CFSTI (TT DOCUMENTS ONLY).

☐ 9.

NBS 9 64

PROCESSOR: & L

COMBINATORIAL PROCESSES AND DYNAMIC PROGRAMMING

Richard Bellman

1. Introduction

The purpose of th~~is~~e paper is to discuss the application of dynamic programming techniques to a class of problems which for want of a better term ~~we~~ are called combinatorial. ~~The essential~~ difficulty of these problems, from the standpoint of the analyst, lies in their apparent lack of complexity. Usually, it is either a question of performing a finite set of arithmetic operations or determining the largest of a finite set of numbers.

If there are one hundred elements in the finite set, we can classify the problem as trivial. If, however, the finite set possesses a million members, or a hundred million, it is worthwhile to ask whether or not there are more efficient techniques than just an element-by-element examination.

Problems of this nature arise in the following ways:

1. Solving linear systems of equations of the form

$$\sum_{j=1}^{N} a_{ij}x_j = b_i, \quad i = 1, 2, \ldots, N.$$

2. Maximization of a linear form $L(x) = \sum_{i=1}^{N} c_i x_i$ subject to constraints of the form

$$\sum_{j=1}^{N} a_{ij}x_j \leq b_i, \quad i = 1, 2, \ldots, M.$$

SUMMARY

The purpose of this paper is to discuss the application of dynamic programming techniques to a class of problems which for want of a better term we call combinatorial. The essential difficulty of these problems, from the standpoint of the analyst, lies in their apparent lack of complexity. Usually, it is either a question of performing a finite set of arithmetic operations or determining the largest of a finite set of numbers.

If there are one hundred elements in the finite set, we can classify the problem as trivial. If, however, the finite set possesses a million members, or a hundred million, it is worthwhile to ask whether or not there are more efficient techniques than just an element-by-element examination.

Problems of this nature arise in the following ways:

1. Solving linear systems of equations of the form

$$\sum_{j=1}^{N} a_{ij} x_j = b_i, \quad i = 1, 2, \ldots, N.$$

2. Maximization of a linear form $L(x) = \sum_{i=1}^{N} c_i x_i$ subject to constraints of the form

$$\sum_{j=1}^{N} a_{ij} x_j \leq b_i, \quad i = 1, 2, \ldots, M.$$

3. Maximization of functions over finite sets, such as permutations, paths along a grid, and so on.

At the present time, there is no systematic theory of problems of this genre, nor is it likely that there ever will be, considering the many varieties and sources. There are, however, some categories of problems recognized as tractable. Some are soluble explicitly in traditional analytic terms, some by means of algorithms that can be carried out by hand, and some require the most powerful computers available.

In what follows, we shall discuss various ways in which the functional equation technique of dynamic programming can be applied. We shall use only those portions of the general theory required for our present purposes, referring the reader interested in further aspects to our book.

Although we shall not present any specific numerical results here, we shall furnish references to extensive computational studies carried out by S. Dreyfus and the author.

CONTENTS

COMBINATORIAL PROCESSES AND DYNAMIC PROGRAMMING

Richard Bellman

1. Introduction

The purpose of this paper is to discuss the application of dynamic programming techniques to a class of problems which for want of a better term we call combinatorial. The essential difficulty of these problems, from the standpoint of the analyst, lies in their apparent lack of complexity. Usually, it is either a question of performing a finite set of arithmetic operations or determining the largest of a finite set of numbers.

If there are one hundred elements in the finite set, we can classify the problem as trivial. If, however, the finite set possesses a million members, or a hundred million, it is worthwhile to ask whether or not there are more efficient techniques than just an element-by-element examination.

Problems of this nature arise in the following ways:

1. Solving linear systems of equations of the form

$$\sum_{j=1}^{N} a_{ij}x_j = b_i, \quad i = 1, 2, \ldots, N.$$

2. Maximization of a linear form $L(x) = \sum_{i=1}^{N} c_i x_i$ subject to constraints of the form

$$\sum_{j=1}^{N} a_{ij}x_j \leq b_i, \quad i = 1, 2, \ldots, M.$$

3. Maximization of functions over finite sets, such as permutations, paths along a grid, and so on.

At the present time, there is no systematic theory of problems of this genre, nor is it likely that there ever will be, considering the many varieties and sources. There are, however, some categories of problems recognized as tractable. Some are soluble explicitly in traditional analytic terms, some by means of algorithms that can be carried out by hand, and some requiring the most powerful computers available.

In what follows, we shall discuss various ways in which the functional equation technique of dynamic programming can be applied. We shall use only those portions of the general theory required for our present purposes, referring the reader interested in further aspects to our book, [1].

Although we shall not present any specific numerical results here, we shall furnish references to extensive computational studies carried out by S. Dreyfus and the author.

2. An Allocation Problem

Let us begin with the following simple allocation problem. Suppose that we have a quantity x of a resource which we are going to subdivide into N parts, $x_1, x_2, \ldots, x_N$, corresponding to N different activities. To make a mathematical problem of this, let us suppose that we are given functions $g_i(x_i)$ which measure the return from the i-th activity due to an allocation x_i.

The question of most efficient allocation of resources

leads to the analytic problem of maximizing the function

(1) $$F_N(x_1, x_2, \ldots, x_N) = g_1(x_1) + g_2(x_2) + \cdots + g_N(x_N)$$

subject to the constraints

(2) (a) $x_1 + x_2 + \cdots + x_N = x,$

(b) $x_i \geq 0.$

Although this may seem like a most prosaic problem, and hardly worth any attention at this late date in the history of calculus, as we shall see it has its hidden pitfalls.

The run-of-the-mill approach to this problem converts it by way of a Lagrange multiplier into that of maximizing the new function

(3) $$G_N(x_1, x_2, \ldots, x_N) = \sum_{i=1}^{N} g_i(x_i) - \lambda \sum_{i=1}^{N} x_i,$$

where λ is a parameter that will be determined from (2a).

The variational equations are

(4) $$\frac{\partial G_N}{\partial x_i} = 0 = g_i'(x_i) - \lambda, \quad i = 1, 2, \ldots, N.$$

Solving these N equations for the x_i in terms of λ, $x_i = x_i(\lambda)$, the parameter λ is determined by means of the relation

(5) $$\sum_{i=1}^{N} x_i(\lambda) = x.$$

Although this approach is infallible in textbook problems, a number of difficulties arise in applications. Let us enumerate them.

In the first place, the functions $g_i(x)$ may not have a derivative. Although, as we shall see below, we do possess an efficient technique for solving a maximization problem of this type, we shall not insist upon this point. It is, however, reasonable to expect that the individual functions need not possess derivatives at various points.

Let us ignore these bizarre possibilities and assume that it is sufficient to examine the solutions of (4) and (5). If each of the functions $g_i'(x_i)$ is monotone, which is to say that $g_i(x)$ is either convex or concave, then the inverse function $x_i(\lambda)$ is uniquely defined and it becomes relatively easy to study the solutions of (5).

Since it is quite common for utility functions $g_i(x)$ to have points of inflection, if we wish to resolve general problems of this nature we must consider situations in which the equations in (4) have a multiplicity of solutions. Assuming, for the sake of moderate complication, that each equation of the form $g_i'(x_i) = \lambda$ possesses two solutions for any particular value of λ, we see that the solution of any equation such as (5) leads to a consideration of 2^N cases.

This number 2^N arises by counting all possible cases. The problem thus appears to have unpleasant combinatorial overtones.

We can amplify these overtones in a number of ways. In the first place, we can insist that the endpoints of the x_i-intervals be tested. The value $x_i = 0$ has a very important interpretation. It means that the i-th activity is not engaged in at all. The problem of taking into account all possibilities of end-point extrema greatly complicates the enumeration of cases.

Secondly, we can impose additional constraints of the type

(6) $$x_i x_{i+1} = 0.$$

The meaning of a constraint of this type is that the use of one activity effectually prevents the use of another.

So far, we have been complaining about the limitations of an approach based on calculus. Let us further curtail this technique by allowing the x_i to range only over the elements of a discrete set. Thus, we may impose the restriction

(7) $$x_i = 0, 1, 2, \ldots, \quad i = 1, 2, \ldots, N.$$

At this point the analyst is tempted to feel that the cards have been too thoroughly stacked against him. Let the computing machines take over; let them solve the problem by the trivial method of examining all possibilities.

The people in charge of the computers, however, may become a bit aggrieved at this attitude. They will concede that they possess fantastic machines operating at phenomenal speeds that can resolve in a matter of hours problems that would have con-

sumed lifetimes even twenty years ago. But these problems must be carefully chosen. Even rudimentary problems of other types cannot be solved by enumeration of cases.

Since this statement may come as a shock to anyone who has not taken the trouble to compute the total number of possible cases arising from simple combinatorial problems, let us illustrate this point by means of a question involving permutations.

Take the problem of placing N objects in N pigeonholes, assuming that we are given a function which measures the value of each assignment of objects. To resolve the problem by examining all cases, we must evaluate N! different cases, corresponding to all different permutations.

Accustomed as we are to the familiar function N!, we seldom realize its rapidity of growth. For N = 10, we have 3,628,000 possibilities, a formidable but not incredible number. For N = 20, it will amuse the reader to calculate how long it would take a computing machine which could evaluate one permutation a microsecond to examine all cases.*

It follows that the mathematician cannot abdicate. He is obligated to develop algorithms which can handle these strange, new problems. Our aim will be to present some simple algorithms which are particularly suited to digital computers. This is not to be considered our ultimate objective, but merely a preliminary step on the way to understanding.

---

*A simple lower bound is half a million years!

3. Functional Equations

Let us now present an approach to these problems quite different from that of the calculus. Introduce the function $f_N(x)$, defined for $x \geq 0$ and $N = 1, 2, \ldots,$ by the relation

$$(1) \qquad f_N(x) = \underset{R_N}{\mathrm{Max}} \left[g_1(x_1) + g_2(x_2) + \cdots + g_N(x_N) \right],$$

where R_N is the region in $(x_1, x_2, \ldots, x_N)$-space defined by the relations

$$(2) \quad (a) \quad x_1 + x_2 + \cdots + x_N = x,$$

$$(b) \quad x_i \geq 0.$$

The only assumption we need make concerning the $g_i(x)$ is that they are continuous for $x_i \geq 0$. In the cases we shall treat below where the x_i assume only a finite set of values, even this restriction will not be necessary.

Let us write*

$$(3) \qquad f_N(x) = \underset{0 \leq x_N \leq x}{\mathrm{Max}} \quad \underset{R_{N-1}(x_N)}{\mathrm{Max}} \left[g_1(x_1) + g_2(x_2) + \cdots + g_N(x_N) \right],$$

where $R_{N-1}(x_N)$ is defined by the inequalities

$$(4) \quad (a) \quad x_1 + x_2 + \cdots + x_{N-1} = x - x_N,$$

---

*This is a particular application of the "principle of optimality, see [1], p. 83.

(b) $x_1 \geq 0$.

Hence, for $N = 2, \ldots,$

(5) $$f_N(x) = \max_{0 \le x_N \le x} \left[g_N(x_N) + \max_{R_{N-1}(x_N)} \left[g_1(x_1) + \cdots + g_{N-1}(x_{N-1}) \right] \right].$$

Thus, referring to the original definition of the sequence $\{f_k(x)\}$,

(6) $$f_N(x) = \max_{0 \le x_N \le x} \left[g_N(x_N) + f_{N-1}(x - x_N) \right], \quad N = 2, 3, \ldots .$$

For $N = 1$, we have

(7) $$f_1(x) = g_1(x).$$

4. Discussion

The preceding formalism reduces the original multidimensional maximization problem to a sequence of one-dimensional problems. The practical significance of this fact is that we now do possess a feasible technique for solving these problems by direct search methods using digital computers.

In this way, we can treat a number of problems arising in mathematical economics, engineering, and operations research. The computational solutions of these questions, joint work with S. Dreyfus, will appear in book form in the near future.

If we attribute some structure to the functions, such as linearity, quadratic character, convexity or concavity, the recurrence relations in (3.5) can be used to determine the analytic character of the sequence $\{f_1(x)\}$ and the maximizing x_1 as functions of x; cf. [1], [2], Karush, [40].

Alternatively, a structural feature such as concavity can be used to accelerate greatly the machine search for a maximum, cf. Kiefer, [41], Johnson, [38], Johnson and Gross, [39], Kiefer, [42]. Applications of the technique are contained in [18], [19].

5. An Imbedding Process

It is important to point out what we have accomplished by means of the functional equation technique. We have taken a particular problem with a specific value of x and N and made it a member of a family of problems, continuous in x and discrete in N.

In other words, we have imbedded a particular process within a family of processes. Oddly, it is easier to treat the particular process by consideration of the whole family of processes, than it is to treat the process by itself.

Further discussion of this point will be found in [1], [2].

6. Constraints

Let us now examine the effect of constraints upon the method outlines in §4. Suppose that we impose the additional constraints

(1) $$0 < a_i \leq x_i \leq b_i, \quad i = 1, 2, \ldots, N,$$

in addition to those of (2.2).

It is easy to see that the relations of (4.5) are replaced by

$$(2) \qquad f_N(x) = \operatorname*{Max}_{S_N}\left[g_N(x_N) + f_{N-1}(x-x_N)\right],$$

where S_N is the x_N-region determined by the new conditions

$$(3) \quad (a) \quad a_N \leq x_N \leq b_N,$$

$$(b) \qquad x_N \leq x - (a_1+a_2+\cdots+a_{N-1}).$$

In the definition of $f_N(x)$, x is restricted by the lower bound $a_1 + a_2 + \cdots + a_N$.

The interesting thing to note is that whereas in the usual approach to maximization problems additional constraints cause difficulties, here the more constraints, the simpler the computational task. Additional constraints restrict the region over which each variable can roam, and thus simplify the search for a maximum. We shall mention this again below.

7. Constraints—Discreteness

As an example of a nastier type of constraint, consider the problem of maximizing

$$(1) \qquad F(x_1,x_2,\ldots,x_N) = \sum_{i=1}^{N} g_i(x_i)$$

subject to the constraints

(2) (a) $x_i = 0$ or 1,

(b) $\sum_{i=1}^{N} x_i \leq x.$

As before, we obtain the recurrence relation

$$f_N(x) = \max_{x_N=0,1} \left[g_N(x_N) + f_{N-1}(x-x_N) \right] \tag{3}$$

$$= \max \left[g_N(1) + f_{N-1}(x-1),\ g_N(0) + f_{N-1}(x) \right].$$

The computation can now be carried out by hand in a very simple fashion. Observe that this is the simplest type of maximization problem that a machine can perform.

8. Mutually Exclusive Activities

Let us now complicate matters still further. In addition to the restrictions in the preceding section, let us impose the constraint

$$x_i x_{i+1} = 0, \quad i = 1, 2, \ldots, N-1. \tag{1}$$

To treat this problem, introduce the sequence of functions of two variables, $f_N(x,y)$, defined by the relations

$$f_N(x,y) = \max_{R_N} \left[g_1(x_1) + g_2(x_2) + \cdots + g_N(x_N) \right], \tag{2}$$

where R_N is now the region in $(x_1, x_2, \ldots, x_N)$-space defined by

(3) (a) $x_1 + x_2 + \cdots + x_N \leq x,$

(b) $x_i = 0, 1,$

(c) $x_i x_{i+1} = 0, \quad i = 1, 2, \ldots, N-1,$

(d) $x_N y = 0.$

The quantity y is allowed to take only two values, 0 or 1.

Then we have the recurrence relation

$$(4) \qquad f_N(x,y) = \operatorname*{Max}_{x_N} \left[g_N(x_N) + f_{N-1}(x - x_N, x_N) \right],$$

where x_N is subject to the conditions

(5) (a) $x_N = 0, 1,$

(b) $x_N \leq x,$

(c) $x_N y = 0.$

In order to resolve the original problem we must compute the two sequences $f_N(x,0)$, $f_N(x,1)$.

It is easily seen that

$$(6) \qquad f_N(x,0) = \operatorname{Max} \left[g_N(1) + f_{N-1}(x-1,1),\ g_N(0) + f_{N-1}(x,0) \right],$$

$$f_N(x,1) = g_N(0) + f_{N-1}(x,0).$$

The two sequences $\{f_N(x,0)\}$, $\{f_N(x,1)\}$, can thus be determined quite simply.

9. More Constraints

Returning to the simpler problem discussed in §2, let us consider the problem of maximizing

$$(1) \qquad F(x_1, x_2, \ldots, x_N) = g_1(x_1) + g_2(x_2) + \cdots + g_N(x_N),$$

subject to the constraints

$$(2) \quad (a) \quad x_1 + x_2 + \cdots + x_N \leq x,$$

$$(b) \quad a_1 x_1 + a_2 x_2 + \cdots + a_N x_N \leq y, \quad a_i \geq 0,$$

$$(c) \quad x_i \geq 0.$$

Observe that we have replaced equality signs by inequalities, since this avoids some unimportant consistency requirements.

Introducing the sequence of functions, $f_N(x,y)$, defined by

$$(3) \qquad f_N(x,y) = \underset{R_N}{\text{Max}}\, F(x_1, x_2, \ldots, x_N)$$

for $N = 1, 2, \ldots,$ $x, y \geq 0$, it is easy to see that, as in the preceding sections, we obtain the recurrence relation

$$(4) \qquad f_N(x,y) = \underset{S_N}{\text{Max}} \left[g_N(x_N) + f_{N-1}(x - x_N, y - y_N) \right],$$

$N \geq 2$, with

$$(5) \qquad f_1(x,y) = \text{Max} \left[g_1(x_1) \right],$$

for $0 \leq x_1 \leq \text{Min}[x, y/a_1]$.

10. General Formulation

There is no difficulty in formulating the problem of maximizing

$$(1) \qquad F_N(x) = \sum_{i=1}^{N} g(x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(k)})$$

subject to the constraints

$$(2) \quad (a) \quad \sum_{i=1}^{N} k_{ij}(x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(k)}) \leq y_j, \quad j = 1, 2, \ldots, M,$$

$$(b) \quad x_i^{(j)} \geq 0,$$

in the same fashion. Setting

$$(3) \qquad f_N(y_1, y_2, \ldots, y_M) = \operatorname*{Max}_{x} F_N(x),$$

we see that

$$(4) \qquad f_N(y_1, y_2, \ldots, y_M) = \operatorname*{Max}_{S_N} \Big[g(x_N^{(1)}, x_N^{(2)}, \ldots, x_N^{(k)}) + f_{N-1}(y_1 - k_{N1}(x_N^{(1)}, \ldots, x_N^{(k)}), \ldots) \Big].$$

Prior to any discussion of the computational feasibility of an algorithm of this type for general values of M, let us turn to a particular problem of this type in which large values of M enter in a most natural way.

11. The Hitchcock-Koopmans Transportation Problem

Let us now discuss one of the most interesting models in mathematical economics, the Hitchcock-Koopmans model of the flow of commodities.

Suppose that at N different locations, which we shall call sources, there are quantities of an item which must be transported to M other locations which we shall call sinks.

Let x_i denote the quantity of the item at the i-th source, y_j denote the demand for this item at the j-th sink, and a_{ij} denote the cost of transporting a unit quantity from the i-th source to the j-th source. Furthermore, assume that the total supply at the sinks is equal to the total demand from the sources.

The problem is to determine a shipping policy which minimizes the cost of supplying the demand. To reduce this problem to analytic form, let

(1) x_{ij} = the quantity sent from the i-th source to the j-th source.

Then we are required to minimize the linear function

$$(2) \qquad L(x) = \sum_{i=1}^{N} \sum_{j=1}^{M} a_{ij} x_{ij}$$

over all x_{ij} satisfying the linear constraints

$$(3) \quad (a) \quad \sum_{j=1}^{N} x_{ij} = x_i,$$

(b) $\sum_{i=1}^{M} x_{ij} = y_j$,

(c) $x_{ij} \geq 0$.

This problem is one that can be treated very successfully by the "simplex technique" of G. Dantzig, [27], or by the newer methods of Fulkerson and Ford, [30]; cf. also Prager, [46]. Both of these methods depend strongly upon the linearity of the various equations.

It can easily be shown that the linearity of all the functions involved prevents the existence of any internal maximum. The region defined by the relations of (3) is the interior of a multi-dimensional polyhedron. To determine the maximum of $L(x)$, it is sufficient to examine the values of $L(x)$ at the vertices of this region.

It follows that we have a problem of combinatorial type. The methods described above furnish efficient search techniques.

These methods fail in general if we introduce nonlinear cost functions. We shall employ functional equation techniques.

12. The Nonlinear Transportation Problem

Let us examine the problem of minimizing

$$(1) \qquad g(x) = \sum_{i=1}^{M} \sum_{j=1}^{N} g_{ij}(x_{ij})$$

over all x_{ij} satisfying the constraints of (10.3), where the $g_{ij}(x)$ are not necessarily linear.

To treat the question by means of functional equations, we can proceed in one of two ways. To begin with, assume that we satisfy the total demand in the following fashion: first, the demand of the M-th sink, then, having satisfied this, the demand of the (M-1)-st sink, and so on.

$$
\begin{array}{lllll}
 & & & 0 & y_1 \\
x_1 & 0 & & 0 & y_2 \\
x_2 & 0 & \searrow x_{1M} & & \\
 & \vdots & \searrow x_{2M} & \vdots & \\
x_N & 0 & \longrightarrow x_{NM} & 0 & y_M .
\end{array}
$$

For fixed demands, y_i, let

(2) $f_M(x_1, x_2, \ldots, x_N)$ = the minimum cost to satisfy the demands of M remaining sinks, starting with quantities $x_1, x_2, \ldots, x_N$ at the N sinks.

Then the same reasoning as we have used above yields the equation

$$
(3) \qquad f_M(x_1, x_2, \ldots, x_N) = \operatorname*{Min}_{x_{iM}} \Big[\sum_{i=1}^{N} g_{iM}(x_{iM}) + f_{M-1}(x_1 - x_{1M}, x_2 - x_{2M}, \ldots, x_N - x_{NM}) \Big]
$$

where the x_{iM} vary over the region determined by

$$(4) \quad (a) \quad \sum_{i=1}^{N} x_{iM} = y_M,$$

$$(b) \quad 0 \leq x_{iM} \leq x_i, \quad i = 1, 2, \ldots, N.$$

The function $f_1(x_1, x_2, \ldots, x_N)$ is given by

$$(5) \qquad f_1(x_1, x_2, \ldots, x_N) = \sum_{i=1}^{N} g_{i1}(x_{i1}).$$

We have thus transformed the original problem into that of computing the sequence $\{f_M(x_1, x_2, \ldots, x_N)\}$.

It is clear that we could obtain an alternative formulation by using first all the resources of the N-th source to satisfy some of the demands at the M sinks, and so on.

13. Feasibility

Let us now see whether or not the recurrence relations presented in (12.3) actually lead to a feasible computational scheme. At each stage of the computation we have to tabulate a function of N variables, and perform a minimization over an N-dimensional region.

Although both of these are formidable procedures if M and N are large, the tabulation problem is at the moment the most difficult. Suppose that we allow each x_i to assume one hundred values, say $x_i = 0, \Delta, \ldots, 99\Delta$. Then the total number of grid-points required to tabulate $f_M(x_1, x_2, \ldots, x_N)$ will be 10^{2N}. If $N = 1$, this is 100, a trivial number; if $N = 2$, this is 10,000, a respectable number; and if $N = 3$, this is 1,000,000, an impossible number at the present time.

It follows that the foregoing method in its straightforward form cannot be used to handle problems of this nature unless N or $M \leq 2$.

Two facts save this from being an academic exercise. In the first place, there are a number of important situations in which N or M is one, two or three. In the second place, as we shall see below, there are a number of devices we can combine with the functional equation technique in order to treat higher dimensional problems.

These are

(1) (a) Lagrange multipliers,

(b) Functional approximation,

(c) Successive approximations.

We shall discuss these ideas in turn.

14. Reduction by One Variable

In view of the tremendous difference between the memory requirements for functions of two variables and functions of three variables, it is of interest to point out that transportation processes involving N sources can be treated by means of functions of $N - 1$ variables. Hence, problems involving two sources are easily resolved, while problems involving three sources can be treated with the best of current machines.

To obtain this reduction in dimensionality, we observe that as yet we have made no use of the fact that supply is equal to demand,

$$(1) \qquad \sum_{i=1}^{N} x_i = \sum_{j=1}^{M} y_j .$$

From this it follows that the values of $x_1, x_2, \ldots, x_{N-1}$ determine the value of x_N, once we have specified the y_i. Hence

$$(2) \qquad f_M(x_1, x_2, \ldots, x_N) = f_M(x_1, x_2, \ldots, x_{N-1}).$$

In much of analysis, dimensionality plays an inessential role. In computational work, it is a basic consideration.

15. Lagrange Multipliers and Dynamic Programming*

We have another very powerful way of evading the curse of dimensionality. Returning to the allocation problem discussed initially, consider the problem of maximizing

$$(1) \qquad F(x_1, x_2, \ldots, x_N) = \sum_{i=1}^{N} g_i(x_i)$$

subject to the constraints

$$(2) \quad (a) \quad \sum_{i=1}^{N} x_i \leq x,$$

$$(b) \quad \sum_{i=1}^{N} a_i x_i = y, \quad a_i \geq 0,$$

$$(c) \quad x_i \geq 0.$$

Observe that we have kept one constraint an equality, one an

*First presented in [3].

inequality. Again there are some reasons of convenience.

As we know, this problem can be treated by means of functions of two variables. However, there is a great incentive for reducing the problem to one that can be handled by functions of one variable.

What we do is combine the functional equation technique with the classical Lagrange multiplier formalism. Consider the problem of maximizing the new function

$$(3) \qquad G(x_1, x_2, \ldots, x_N) = \sum_{i=1}^{N} g_i(x_i) - \lambda \sum_{i=1}^{N} a_i x_i$$

subject to the constraints

$$(4) \quad (a) \quad \sum_{i=1}^{N} x_i \leq x,$$

$$(b) \quad x_i \geq 0,$$

where λ is an as yet undetermined parameter.

For fixed λ, introduce the sequence of functions

$$(5) \qquad f_N(x) = \underset{R_N}{\operatorname{Max}} \left[\sum_{i=1}^{N} g_i(x_i) - \lambda \sum_{i=1}^{N} a_i x_i \right],$$

where R_N is defined only by (4). Then, as before, we readily compute the sequence $\{f_k(x)\}$ by means of the relations

$$(6) \qquad f_N(x) = \underset{0 \leq x_N \leq x}{\operatorname{Max}} \left[g_N(x_N) - \lambda x_N + f_{N-1}(x - x_N) \right].$$

Let $x_1(\lambda)$, $x_2(\lambda)$, ..., $x_N(\lambda)$ be a set of values yielding

the maximum of $G(x_1, x_2, \ldots, x_N)$. Then we assert that these values yield the solution to the problem of maximizing (1) subject to the constraint in (2) where y is determined by

$$y = \sum_{i=1}^{N} a_i x_i(\lambda). \tag{7}$$

To prove this, proceed by contradiction. Suppose that there existed values $(z_1, z_2, \ldots, z_N)$ satisfying (2) such that

$$F(z_1, z_2, \ldots, z_N) > F(x_1(\lambda), x_2(\lambda), \ldots, x_N(\lambda)). \tag{8}$$

Then

$$\begin{aligned} F(z_1, z_2, \ldots, z_N) - \lambda \sum_{i=1}^{N} a_i z_i &= F(z_1, z_2, \ldots, z_N) - \lambda y \\ &> F(x_1(\lambda), x_2(\lambda), \ldots, x_N(\lambda)) - \lambda y \\ &= F(x_1(\lambda), x_2(\lambda), \ldots, x_N(\lambda)) - \lambda \sum_{i=1}^{N} a_i x_i(\lambda). \end{aligned} \tag{9}$$

This, however, yields a contradiction, since the $x_i(\lambda)$ were obtained as a solution to (5), subject to the constraints of (4).

Although there is no difficulty in letting the results justify the method in any particular application, there are a number of important facts which remain to be verified. We suspect that as λ varies from $-\infty$ to $+\infty$ that the value of $\sum_{i=1}^{N} a_i x_i(\lambda)$ will vary between its maximum and minimum, and, furthermore that this variation will be monotone and continuous.

The monotonicity is not only of theoretical importance, but of practical significance in determining the value of λ for which $\sum_{i=1}^{N} a_i x_i(\lambda) = y$; cf. Gross and Johnson, [3]. Some applications of this technique will be found in [17], [18].

16. Discussion

The importance of the procedure outlined above resides in the fact that it enables us to partition a problem originally requiring a sequence of functions of two variables into a sequence of problems requiring functions of one variable.

There is no difficulty in extending these techniques to treat the case where there are M constraints. What we gain in reducing dimensionality on one hand, we must pay for in multi-dimensional search on the other.

As we know, the introduction of Lagrange multipliers is equivalent to introducing dual variables; cf. Kuhn and Tucker, [4]. What we have done above is to operate partially in the original space and partially in the dual space; partly in the space of "resources" and partly in the space of "prices."

17. Functional Approximation

In the previous sections, when we have discussed the computational solution of functional equations, we have tacitly equated the concept of a function $f(x)$ defined over an interval $[0,a]$ with a set of values $\{f(k\Delta)\}$, where $N\Delta = a$, and Δ is some grid size. The finer the grid, the more values that must be computed. Similarly, a function of two variables, $f(x,y)$, is equivalent to a sequence of values $\{f(k\Delta, \ell\Delta)\}$.

As we increase the number of independent variables, the number of grid-points goes up at an exponential rate. It is this fact that defeats the effective use of the algorithms presented above in a number of significant processes.

It follows that one way to defeat this exponential growth in the information required to specify a function is to use a different description.

Consider, for example, a power series expansion

$$(1) \qquad f(x) = \sum_{n=0}^{\infty} a_n x^n,$$

convergent for $0 \le x \le a$. If we truncate the series and use the polynomial $\sum_{n=0}^{N} a_n x^n$ as an approximation to the function, we see that $f(x)$ is determined for all x in $[0,a]$ by the $N + 1$ coefficients, a_i, $i = 0, 1, \ldots, N$, and thus by $(N+1)$ quantities.

Power series expansions have the drawback of being associated with analyticity and, in addition, of not providing uniformly good approximation over the entire interval. Let us then use instead an orthonormal expansion

$$(2) \qquad f(x) \sim \sum_{n=0}^{\infty} a_n \phi_n(x)$$

where the functions $\phi_i(x)$ are elements of a complete orthonormal system. For a finite interval, two particularly important choices are those of trigonometric functions, $\{\sin kx, \cos kx\}$, and of Legendre polynomials.

We write

$$(3) \qquad f(x) = \sum_{n=0}^{N} a_n \phi_n(x),$$

where

$$(4) \qquad a_n = \int_0^a f(x)\phi_n(x)dx.$$

In evaluating this integral, we don't wish to use a Riemann sum, say

$$(5) \qquad a_n \cong \sum_{k=0}^{M} f(k\Delta)\phi_n(k\Delta)\Delta,$$

since this will involve the calculation of $f(k\Delta)$ for all k, precisely the type of computation we wished to avoid.

Consequently, we employ a numerical integration formula of the form

$$(6) \qquad \int_0^a g(x)dx \cong \sum_{i=1}^{M} c_i g(x_i),$$

where x_i are fixed points in $[0,a]$, independent of $g(x)$ but dependent on M, and the c_i are likewise fixed coefficients independent of $g(x)$, but dependent on M.

Thus

$$(7) \qquad a_n \cong \sum_{i=1}^{M} c_i f(x_i)\phi_n(x_i) \cong \sum_{i=1}^{M} d_{in} f(x_i),$$

since the quantities $\phi_n(x_i)$ can be calculated once and for all.

Observe the interesting fact about this formula that the

value of a_n, and thus of $f(x)$ is made to depend upon the values of $f(x_i)$ at a fixed set of points $\{x_i\}$.

Let us see then how the calculation proceeds. Turning to the recurrence relation

$$(8) \qquad f_N(x) = \max_{0 \le x_N \le x} \left[g_N(x_N) + f_{N-1}(x-x_N) \right],$$

suppose that x is restricted to an interval $[0,a]$. Starting with the function $f_1(x) = g_1(x)$, we reduce $f_1(x)$ to a sequence of coefficients

$$(9) \qquad f_1(x) \sim \left[a_0^{(1)}, a_1^{(1)}, \ldots, a_N^{(1)} \right].$$

We see from the above discussion that to determine $f_2(x)$, we need only calculate the values $f_2(x_i)$. Hence, we compute these from the relations

$$(10) \qquad f_2(x_i) = \max_{0 \le x \le x_i} \left[g_N(x_2) + f_1(x_i-x_2) \right].$$

The values of $f_1(x_i-x_2)$ are determined from (7).

Having computed $f_2(x_i)$, $i = 1, 2, \ldots, M$, in this fashion, we determine the new coefficients $a_i^{(2)}$ using (7). The function $f_2(x)$ is thus reduced to a sequence of coefficients

$$(11) \qquad f_2(x) \sim \left[a_0^{(2)}, a_1^{(2)}, \ldots, a_N^{(2)} \right].$$

We now repeat this process to determine as many elements of the sequence $\{f_N(x)\}$ as desired.

A number of questions remain before this technique can be applied. We must determine N in (3) and M in (7), and the type of orthonormal sequence. The choice of N and M depend upon the accuracy desired and the facilities available.

Both the trigonometric functions, $\{\sin Nx, \cos Nx\}$, and the Legendre polynomials possess simple recurrence relations which permit the N-th member of the sequence to be computed from the values of the first members.

As far as quadratic formulas are concerned, it is probably best to use Gaussian quadrature, which, as we know, is exact for polynomials up to degree $2M - 1$ if M points are used in (6). For an application of this technique, see [1[illegible]].

18. Cebycev Approximation

The approximation

$$(1) \qquad f(x) \cong \sum_{n=0}^{N} a_n \phi_n(x)$$

is equivalent to choosing the coefficients $\{a_i\}$ according to mean-square approximation. If the b_i are determined so as to minimize the mean-square deviation

$$(2) \qquad \int_0^a \left[f(x) - \sum_{n=0}^{N} b_n \phi_n(x) \right]^2 dx,$$

we find that $b_i = a_i$.

However, mean-square deviation is less desirable than Cebycev approximation,

$$(3) \qquad \underset{b_i}{\mathrm{Min}} \ \underset{0 \leq x \leq a}{\mathrm{Max}} \left| f(x) - \sum_{n=0}^{N} b_n \phi_n(x) \right| .$$

Unfortunately, no simple representation for the minimizing b_1, corresponding to (17.4) exists. Nevertheless, there are available feasible computational techniques for determining the minimizing b_1 in (3).

19. Functions of Several Variables

In the previous sections, we have shown how a function defined over $[0,a]$ may be described by a relatively small set of parameters. The same process can be applied to a function of two variables, $f(x,y)$, defined over $0 \leq x,y \leq a$,

$$(1) \qquad f(x,y) \cong \sum_{m,n=0}^{N} a_{mn}\phi_m(x)\phi_n(y).$$

We see that functions of two variables will require $(N+1)(N+2)/2$ coefficients, while those of three variables will require roughly $N^3/6$ coefficients.

Take $N = 10$, we have functions of two variables determined by 66 quantities, and functions of three variables determined by approximately 200 quantities. These numbers compare very favorably with 10^4 and 10^6 arising from $10^2 \times 10^2$ grids and $10^2 \times 10^2 \times 10^2$ grids.

In any particular problem, a certain amount of experimentation will be required.

Again an important point to stress is that these techniques allow us to treat problems which cannot be treated by straightforward tabulation of functional values at grid-points.

20. Successive Approximations

Let us now discuss an approach of entirely different type to the problem of solving multi-dimensional problems in terms of functions of a small number of variables. We wish to employ the classical method of successive approximations.

To illustrate the workings of the method, let us give two examples of its use, one in connection with the allocation problem described above, and one in connection with the Hitchcock-Koopmans problem.

Consider the problem of maximizing

(1) $$h(x_1,\dots,x_N;y_1,\dots,y_N) = \sum_{k=1}^{N} g_k(x_k,y_k)$$

subject to the constraints

(2) (a) $$\sum_{k=1}^{N} x_k = x, \quad x_k \geq 0,$$

(b) $$\sum_{k=1}^{N} y_k = y, \quad y_k \geq 0.$$

As we know, this problem can be treated by means of sequences of functions of two variables, §9, and by means of sequences of functions of one variable using Lagrange multiplier techniques. Let us now treat it by means of successive approximations.

Let $(y_1^{(0)},y_2^{(0)},\dots,y_N^{(0)})$ be an initial guess concerning the choice of the y_1 and consider the problem of maximizing the function

$$(3) \qquad h(x_1,\ldots,x_N;y_1^{(0)},\ldots,y_N^{(0)}) = \sum_{k=1}^{N} g_k(x_k,y_k^{(0)})$$

subject to the constraint of (2a). This problem can, as we know, be resolved via functions of one variable.

Call a maximizing set of x_k $\{x_k^{(0)}\}$. Now consider the problem of maximizing

$$(4) \qquad h(x_1^{(0)},\ldots,x_N^{(0)};y_1,\ldots,y_N) = \sum_{k=1}^{N} g_k(x_k^{(0)},y_k),$$

subject to the constraints of (2b). This again is a one-dimensional problem in our terms. Call a maximizing set $\{y_k^{(1)}\}$. The pattern of procedure is now set. We obtain alternately maximizing sequences $\{y_k^{(i)}\}$ and $\{x_k^{(i)}\}$, with corresponding approximations to the desired maximum value,

$$(5) \qquad h(x_1^{(i)},\ldots,x_N^{(i)};y_1^{(i)},\ldots,y_N^{(i)}),$$

$$h(x_1^{(i)},\ldots,x_N^{(i)};y_1^{(i+1)},\ldots,y_N^{(i+1)}).$$

We have thus once again reduced a problem originally requiring functions of two variables to one requiring sequences of functions of one variable.

21. Monotonicity of Approximation

Let

$$(1) \qquad u_{2i+1}(x,y) = \operatorname*{Max}_{x} h(x,y^{(i)}), \quad i = 0,\ 1,\ \ldots,$$

$$u_{2i}(x,y) = \operatorname*{Max}_{y} h(x^{(i)},y), \quad i = 1,\ 2,\ \ldots\ .$$

It is clear that

$$(2) \qquad u_1(x,y) \leq u_2(x,y) \leq \cdots .$$

Hence, the sequence $\{u_k(x,y)\}$ converges monotonically. It is, however, not clear that it converges to the absolute maximum. This requires a separate discussion which we shall present elsewhere.

22. Approximation in Policy Space

This monotonicity of approximation is not accidental. The type of approximation we have been employing is a particular type of approximation in "policy space," which necessarily yields monotonicity.

In place of approximating to the return functions, the $f_k(x,y)$ as defined in §9, the usual method of successive approximation, we operate partially in the space of policy functions.

For a further discussion of approximation in policy space, see [1], Chapter 3, and for some further applications of successive approximations, see [4], [24].

23. Application to Hitchcock-Koopmans Transportation Problem

One way of applying these ideas to either the linear or nonlinear transportation is the following. Let the shipments from the 3-rd to N-th source be assigned arbitrarily, and consider the problem of determining the shipments from the first two sources which will minimize the cost of supplying the remaining demand.

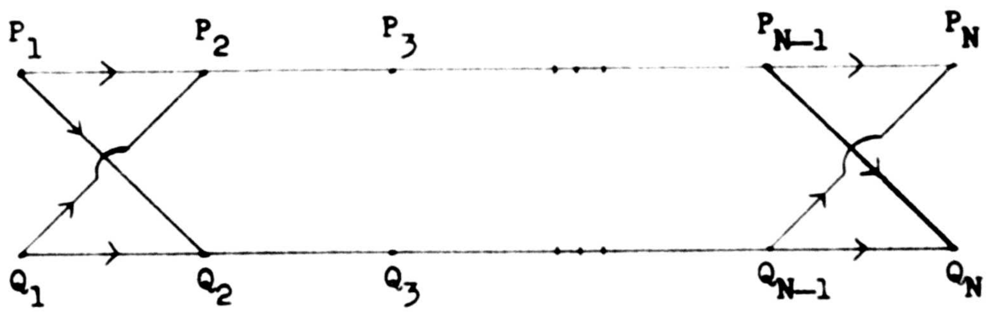
This, as we know, can be done using functions of one variable. Having obtained the optimal shipments from the first and second sources, we use this shipping policy from the first source, retain the shipments previously used from the fourth to N-th source, and consider the new problem of determining the shipments from the second and third sources which will minimize the cost of supplying the remaining demand.

Continuing in this way, it is clear that we obtain a sequence of costs which decrease monotonically and thus converge. Again it is necessary to determine when there is convergence to the actual minimum.

The process can be speeded up, by considering three sources at a time and one Lagrange multiplier to retain the use of functions of one variable.

24. The Harris Transportation Problem

Consider the following network which can be used to describe certain types of flow of information or flow of commodities.



As the arrows indicate, the only permissible flows are from P_i to P_{i+1}, P_i to Q_{i+1}, Q_i to P_{i+1}, and Q_i to Q_{i+1}.

We introduce quantities, which we can call <u>capacities</u>, defined as follows:

(1) a_i = maximum rate of flow over the link between P_i and P_{i+1},
b_i = the same quantity for P_i and Q_{i+1},
c_i = the same quantity for Q_i and P_{i+1},
d_i = the same quantity for Q_i and Q_{i+1}.

Assuming that we are given fixed rates of flow, x and y, into P_1 and Q_1, we wish to maximize a prescribed function of the rate of flow into P_N and Q_N. At each of the terminals, we have a choice of dividing the input flows into two output flows.

To treat this problem, we introduce the sequence of functions,

(2) $f_i(x,y)$ = the maximum of the prescribed function of the flow into P_N and Q_N, given rates of flow x and y into P_i and Q_i respectively,

for $i = 1, 2, \ldots, N-1$, $x, y \geq 0$.

Then, as above, we obtain the recurrence relations

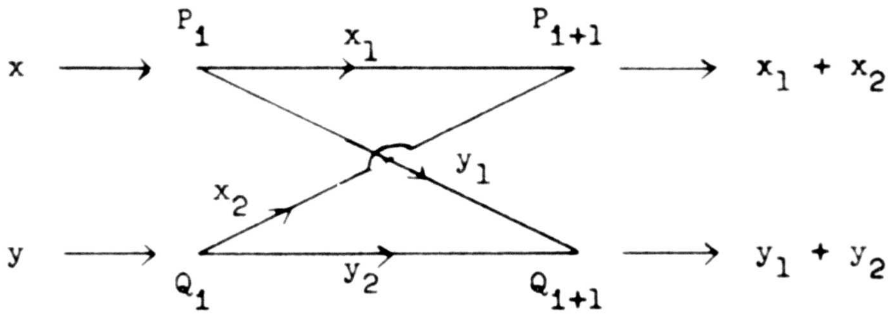
$$(3) \qquad f_i(x,y) = \operatorname*{Max}_{R_i} f_{i+1}(x_1+x_2, y_1+y_2),$$

where $R_i = R_i(x_1, x_2, y_1, y_2)$ is determined by the constraints

(4) (a) $x_1 + y_1 \leq x, \quad x_2 + y_2 \leq y,$

(b) $0 \leq x_1 \leq a_i, \quad 0 \leq x_2 \leq b_i,$

(c) $0 \leq y_1 \leq c_i, \quad 0 \leq y_2 \leq d_i.$



25. General Network Problems

Networks of general type, where the points are irregularly arranged with great freedom of intercommunication cannot be treated in the foregoing fashion. The problem of determining optimal flow over systems of this type was first posed by Harris, [34], and has given rise to a good deal of research.

The general problem has been most successfully attacked by Ford and Fulkerson, [30], who have developed techniques for the treatment of this specific process which have proved to have much wider applicability.

On the other hand, stimulated by the same process, Boldyreff, [24], has developed a very interesting and flexible relaxation technique, the "flooding technique," which also has a wide range of applications.

26. A Routing Problem

Let us now consider the following problem. Suppose that we have a set of N points in the plane or in space, with the property that every two points, P_i and P_j, have an associated number, d_{ij}, which we can call the time required to travel from P_i to P_j. It is not necessary to assume that $d_{ij} = d_{ji}$.

To take account of the fact that in any particular situation two points may not be mutually accessible, we can let $d_{ij} = \infty$.

Given the matrix (d_{ij}), where $d_{ii} = 0$, the problem is to trace a route of minimum time from P_1 to P_N.

To treat this, introduce the $N - 1$ quantities, f_i, defined by

(1) $$f_i = \text{the minimum time to travel from } P_i \text{ to } P_N.$$

It is easy to see that

(2) $$f_i = \underset{j \neq i}{\text{Min}} \left[d_{ij} + f_j \right].$$

Since this system of non-linear equations does not determine the sequence $\{f_i\}$ recurrently, we must use some method of successive approximations to obtain the f_i.

Perhaps the simplest is one based upon approximation in policy space. Let

(3) $$f_i^{(0)} = d_{iN}, \quad i = 1, 2, \ldots, N - 1,$$

and

$$(4) \qquad f_1^{(k)} = \min_{j \neq 1} \left[d_{1j} + f_j^{(k-1)} \right],$$

for $k = 1, 2, \ldots$.

Since this process corresponds to examining first all direct routes from 1 to N, then all that make one step, and so on, it is clear that we will have monotone decreasing convergence. Further discussion concerning the equations will be found in [5].

27. The Traveling Salesman Problem

As an example of a problem of closely related type where the direct functional equation technique fails, consider the problem of drawing a polygonal path of minimum length through N given points in the plane, $P_1, P_2, \ldots, P_N$.

It is clear that the principle of optimality is still valid. No matter what part of the path has already been traversed, the remainder of the path must be of minimum length. What prevents a simple application of recurrence relations is the fact that we must keep track of where we have been. The problem thus has certain features in common with a variety of "excluded volume" problems arising in mathematical physics.

Furthermore, it is a very nice example of the virtual impossibility of gauging the level of difficulty of a simply stated combinatorial problem. Recently, linear programming techniques plus ingenuity have proved successful in solving particular questions of this nature, see Dantzig and Johnson, [28].

To treat this problem by means of functional equations and

successive approximations, let us begin with the question of tracing a path through N given points which must start at P_1 and end at P_N.

Introduce to this end the sequence of functions

(1) $f(Q_1, Q_2, \ldots, Q_k)$ = the minimum length of the remaining path from Q_k to P_N, having been through $Q_1, Q_2, \ldots, Q_k$.

Here the Q_i are particular elements of the P_j-set.

It is easy to see that we obtain the following relations:

(2)
$$f(P_1) = \operatorname*{Min}_{j \neq 1} \left[d_{1j} + f(P_1, P_j) \right],$$

$$f(P_1, P_j) = \operatorname*{Min}_{k \neq j} \left[d_{jk} + f(P_1, P_j, P_k) \right],$$

and so on.

It remains to discuss the computational feasibility of a solution based upon this algorithm. The tabulation of $f(P_1, P_j)$ requires $(N-1)$ values; that of $f(P, P_j, P_k)$ requires $(N-1)(N-2)/2$. What is of great help is the fact that the order of Q_i in (1) is of no importance, so that the tabulation of $f(Q_1, Q_2, \ldots, Q_k)$ requires $(N-1)(N-2)\cdots(N-k)/k!$ entries, rather than $(N-1)(N-2)\cdots(N-k)$.

The maximum number of entries will be required when $k = [N/2]$. To illustrate the order of magnitude of these quantities, we have

(3) $10C_5 = 252, \quad 16C_8 = 12{,}970.$

It follows that with current machines it would be possible to solve problems of this type in a direct fashion for $N \leq 17$.

Let us note in passing that the same technique can be applied to problems of optimal trajectory arising in rocket problems, and in the study of general variational processes; cf. Cartaino and Dreyfus, [25], and [6], [1].

28. Successive Approximations

To treat problems of larger magnitude, we can combine functional equation techniques with the method of successive approximations.

Let $P_1Q_2Q_3 \dots P_N$ be a proposed route. To test this, let us examine the sub-route $P_1Q_2Q_3 \dots Q_9$, and pose the problem of pursuing a path from P_1 to Q_9, going through the points $Q_2, Q_3, \dots, Q_8$, and of minimal length.

This problem can easily be resolved computationally, using the recurrence relation method of the preceding section. Let $P_1R_2R_3 \dots Q_9$ be the new path of minimum length through these ten points, and let us test in this way the set of points $R_5R_6R_7R_8Q_9P_{10}P_{11}P_{12}P_{13}P_{14}$. Continuing in this way we obtain a monotone sequence of decreasing lengths. Again it will be necessary to study whether or not this scheme of successive approximations converges to the absolute minimum.

29. A Class of Scheduling Problems—The Book-binding Problem

Let us now turn to another class of problems which require maximization over permutation.

Suppose that we have N manuscripts which must be printed and bound, in that order, before being published. Suppose further that we have one printing press and one binding machine. Given the quantities

(1) (a) a_i = the time required to print the i-th book,

(b) b_i = the time required to bind the i-th book,

the problem is to determine the order in which the N manuscripts should be processed so as to minimize the total time required for their printing and binding.

This problem is representative of a large class of questions of this nature which arise in scheduling theory. A very simple solution of this problem was given by Johnson, [37], and a derivation of this solution by functional equation techniques was given in [7].

If we add a third operation, say typing, the problem appears to enter the hopeless case. At the present time, we do not even possess any reasonable approximate policies.

30. The Caterer Problem

It is possible, in a number of cases, to reduce scheduling problems to maximization problems of the type encountered in the Hitchcock-Koopmans transportation problem. Having done this, we can introduce functional equations by various artifices. One example of this is in connection with "warehousing" problems, cf. [9], Dreyfus, [29], and, also, [13]. Here, we shall discuss another example, the "caterer" problem.

Let us state the version of this problem given by Jacobs, [35], or Prager, [46].

"A caterer knows that in connection with the meals he has arranged to serve during the next n days, he will need r_j fresh napkins on the j-th day, for $j = 1, 2, \ldots, n$. There are two types of laundry service available. One type requires p days and costs b cents per napkin; a faster service requires q days, $q < p$, but costs c cents per napkin, $c > b$. Beginning with no usable napkins on hand or in the laundry, the caterer meets the demands by purchasing napkins at a cents per napkin. How does the caterer purchase and launder napkins so as to minimize the total cost for n days?"

This problem can be resolved by linear programming techniques in some cases, see the above references and also Gaddum, Hoffman, and Sokolowsky, [32].

What is interesting about this problem from the standpoint of dynamic programming and functional equations is that it appears upon first glance to be a problem requiring functions of q variables. As it turns out, however, the linearity of the process permits us to make a certain type of preliminary transformation which reveals the true dimensionality of the problem. Surprisingly, this is $p - q$. The same type of transformation has proved of great service in connection with a number of variational problems arising in the theory of control processes and elsewhere, see [4], [10].

In the present case, these transformations reduce the problem to that of determining the maximum of the linear form

(1) $$L(v) = v_1 + v_2 + \cdots + v_n$$

subject to the constraints

(2) (a) $r_1 \geq v_1 \geq 0,$

(b) $v_1 \leq b_1,$

$v_1 + v_2 \leq b_2,$

$\vdots$

$v_1 + v_2 + \cdots + v_k \leq b_k,$

$v_2 + v_3 + \cdots + v_{k+1} \leq b_{k+1},$

$\vdots$

$v_{n-k+1} + \cdots + v_n \leq b_n.$

A further surprising fact about this problem is that we can exhibit an explicit analytic solution, a property that is quite rare in this domain.

31. Bottleneck Problems

Let us, without going into any detail, mention a class of problems which we have called "bottleneck" problems because the operation of the system depends at each stage upon the scarcest resources.

The general question is that utilizing a complex of interdependant industries to produce one or two essential items. Using a "lumped" model of economic interaction we consider the state of the system at any time to be specified by a capacity

vector $x(t)$ and a stockpile vector $y(t)$.

Considering first a continuous process, since these are usually more amenable to solution, we meet the problem of determining vectors $z(t)$ and $w(t)$ so as to maximize the inner product

(1) $$I(x,y) = (x(T),a) + (y(T),b),$$

where x and y are determined by linear equations

$$\frac{dx}{dt} = A_1x + B_1y + C_1z + D_1w, \quad x(0) = c,$$

$$\frac{dy}{dt} = A_2x + B_2y + C_2z + D_2w, \quad y(0) = d,$$

under appropriate proportionality assumptions, and the vectors z and w are subject to further feasibility constraints

(3) $$Ez + Fw \leq G + Hx + Iy$$

for $0 \leq t \leq T$.

The novel features of the problem are introduced by the combination of linear equations and linear constraints. The continuous version can be resolved explicitly in a number of cases, see [1], [11]. In addition, Lehman has devised a continuous version of the simplex technique which seems quite promising, [4].

The discrete version can be simply treated by computational techniques if the number of state variables does not exceed 3. A transformation of the problem enables us to reduce the number of variables by one, and simultaneously to keep all variables

within a uniformly bounded region; cf. [1], [15].

If the constraints in (3) are of the simpler form

$$(4) \qquad Ez + Fw \leq G,$$

the techniques sketched in [10] can be used to reduce drastically the dimensionality of the problem.

A detailed discussion of problems of this type, together with solutions of some special cases is given in [1]. Some further results are found in Lehman, [45].

32. Slightly Intertwined Matrices

The functional equation technique is designed to take advantage of structural features of the process. In the preceding discussion we have utilized the multistage aspects in a very natural way. Let us now indicate another way in which functional equations can be employed. These results were presented in [12].

The general linear programming problem is that of maximizing a linear form $L(x) = (x,a)$ subject to linear constraints of the form $Ax \leq b$.

If A is block-diagonal,

$$(1) \qquad A = \begin{pmatrix} A_1 & & & \\ & A_2 & & \\ & & \cdots & \\ & & & A_r \end{pmatrix}$$

the problem breaks up into a set of r independent problems of

smaller dimension. Once again we are concerned with dimensionality in connection with computational feasibility.

Similarly, if A is block-triangular, which is to say that there are zeroes above or below the diagonal matrices, the problem reduces to a sequence of problems of lesser difficulty.

Here we wish to show how functional equation techniques can be utilized to treat a class of problems in which A has approximately a block-diagonal form.

Specifically, let us consider the question of determining the maximum of

$$L_N(x) = \sum_{i=1}^{3N} x_i, \tag{2}$$

subject to the constraints

$$
\begin{aligned}
&a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \leq c_1,\\
&a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \leq c_2,\\
&a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + b_1x_4 \leq c_3,\\
&a_{44}x_4 + a_{45}x_5 + a_{46}x_6 \leq c_4,\\
&a_{54}x_4 + a_{55}x_5 + a_{56}x_6 \leq c_5,\\
&a_{64}x_4 + a_{65}x_5 + a_{66}x_6 + b_2x_7 \leq c_6,\\
&\quad\vdots\\
&a_{3N-2,3N-2}x_{3N-2} + a_{3N-2,3N-1}x_{3N-1} + a_{3N-2,3N}x_{3N} \leq c_{3N-2},\\
&a_{3N-1,3N-2}x_{3N-2} + a_{3N-1,3N-1}x_{3N-1} + a_{3N-1,3N}x_{3N} \leq c_{3N-1},\\
&a_{3N,3N-2}x_{3N-2} + a_{3N,3N-1}x_{3N-1} + a_{3N,3N}x_{3N} \leq c_{3N},
\end{aligned}
\tag{3}
$$

and

(4) $\qquad x_i \geq 0.$

It is assumed that $a_{ij} \geq 0$, $b_i > 0$, with sufficiently many $a_{ij} > 0$ so that the maximum is not infinite.

Let us now define a sequence of functions of z,

(5) $$f_N(z) = \operatorname*{Max}_{x_i} L_N(x),$$

where the x_i are subject to the constraints given above, with the exception that the last constraint is now

(6) $$a_{3N,3N-2}x_{3N-2} + a_{3N,3N-1}x_{3N-1} + a_{3N,3N}x_{3N} \leq z.$$

Employing the principle of optimality, we see that the sequence $\{f_N(z)\}$ satisfies the recurrence relation

(7) $$f_N(z) = \operatorname*{Max}_{[x_{3N-2},x_{3N-1},x_{3N}]} \Big[x_{3N-2} + x_{3N-1} + x_{3N} + f_{N-1}(c_{3N-3} - b_{N-1}x_{3N-2}) \Big],$$

$$N \geq 1,$$

with the variables x_{3N-2}, x_{3N-1}, x_{3N} subject to the constraints

$$(8)\qquad \begin{aligned}
&a_{3N-2,3N-2}x_{3N-2} + a_{3N-2,3N-1}x_{3N-1} + a_{3N-2,3N}x_{3N} \le c_{3N-2},\\
&a_{3N-1,3N-2}x_{3N-2} + a_{3N-1,3N-1}x_{3N-1} + a_{3N-1,3N}x_{3N} \le c_{3N-1},\\
&a_{3N,3N-2}x_{3N-2} + a_{3N,3N-1}x_{3N-1} + a_{3N,3N}x_{3N} \le z,\\
&b_{N-1}x_{3N-2} \le c_{3N-3},\\
&x_{3N-2}, x_{3N-1}, x_{3N} \ge 0.
\end{aligned}$$

The function $f_0(z)$ is identically zero.

33. Reduction in Dimensionality

Let us write the recurrence relation of (32.7) in the form

$$(1)\qquad \begin{aligned}
f_N(z) &= \operatorname*{Max}_{x_{3N-2}} \left[\operatorname*{Max}_{x_{3N-1},x_{3N}} [\cdots] \right]\\
&= \operatorname*{Max}_{x_{3N-2}} \left[\operatorname*{Max}_{R_N} (x_{3N-1}+x_{3N}) + x_{3N-2} \right.\\
&\qquad\qquad \left. + f_{N-1}(c_{3N-3}-b_{N-1}x_{3N-2}) \right]
\end{aligned}$$

where R_N is the region in (x_{3N-1}, x_{3N}) space defined by

$$(2)\qquad \begin{aligned}
&a_{3N-2,3N-1}x_{3N-1} + a_{3N-2,3N}x_{3N} \le c_{3N-2} - a_{3N-2,3N-2}x_{3N-2},\\
&a_{3N-1,3N-1}x_{3N-1} + a_{3N-1,3N}x_{3N} \le c_{3N-1} - a_{3N-1,3N-2}x_{3N-2},\\
&a_{3N,3N-1}x_{3N-1} + a_{3N,3N}x_{3N} \le z - a_{3N,3N-2}x_{3N-2},\\
&x_{3N-1}, x_{3N} \ge 0.
\end{aligned}$$

Thus we can write

$$(3)\qquad f_N(z) = \operatorname*{Max}_{x_{3N-2}} \left[g_N(x_{3N-2}, z) + f_{N-1}(c_{3N-3}-b_{N-1}x_{3N-2}) \right]$$

where x_{3N-2} is constrained by

$$(4)\quad 0 \le x_{3N-2} \le \mathrm{Min}\left[\frac{c_{3N-3}}{b_{N-1}}, \frac{c_{3N-2}}{a_{3N-2,3N-2}}, \frac{c_{3N-1}}{a_{3N-1,3N-2}}, \frac{z}{a_{3N,3N-2}}\right].$$

The function $g_N(y,z)$ is readily determined, since the maximum over R_N is attained at a vertex of the region.

34. Slightly Intertwined Symmetric Matrices

Let us now consider the problem of resolving a set of linear equations of the forms

$$
\begin{aligned}
(1)\qquad & a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = c_1,\\
& a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = c_2,\\
& a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + b_1x_4 = c_3,\\
b_1x_3 + {} & a_{44}x_4 + a_{45}x_5 + a_{46}x_6 = c_4,\\
& a_{54}x_4 + a_{55}x_5 + a_{56}x_6 = c_5,\\
& a_{64}x_4 + a_{65}x_5 + a_{66}x_6 + b_2x_7 = c_6,\\
& \qquad\vdots\\
b_{N-1}x_{3N-3} + {} & a_{3N-2,3N-2}x_{3N-2} + a_{3N-2,3N-1}x_{3N-1} + a_{3N-2,3N}x_{3N} = c_{3N-2},\\
& a_{3N-1,3N-2}x_{3N-2} + a_{3N-1,3N-1}x_{3N-1} + a_{3N-1,3N}x_{3N} = c_{3N-1},\\
& a_{3N,3N-2}x_{3N-2} + a_{3N,3N-1}x_{3N-1} + a_{3N,3N}x_{3N} = c_{3N}.
\end{aligned}
$$

A matrix of the type appearing above, we shall call <u>slightly intertwined</u>. It arises in a variety of physical, engineering, and economic problems involving multicomponent systems with weak coupling.

Let us introduce the matrices

(2) $$A_k = (a_{i+3k-3,\, j+3k-3}), \quad i, j = 1, 2, 3,$$

for $k = 1, 2, \ldots, N$, and the vectors

(3) $$x^k = (x_{3k-2}, x_{3k-1}, x_{3k}), \quad c^k = (c_{3k-2}, c_{3k-1}, c_{3k}).$$

Since the matrix of coefficients is, by assumption, positive definite, the solution of the linear system (1) is equivalent to determining the minimum of the inhomogeneous quadratic form

(4) $$\begin{aligned} &(x^1, A_1 x^1) + (x^2, A_2 x^2) + \cdots + (x^N, A_N x^N) \\ &\quad - 2(c^1, x^1) - 2(c^2, x^2) - \cdots - 2(c^N, x^N) \\ &\quad + 2b_1 x_3 x_4 + 2b_2 x_6 x_7 + \cdots + 2b_{N-1} x_{3N-3} x_{3N-2}. \end{aligned}$$

For $N = 1, 2, \ldots,$ and $-\infty < z < \infty$, let us introduce the sequence of functions of the variable z defined by

(5) $$\begin{aligned} f_N(z) = \operatorname{Min}_{x_i} \Big[&\sum_{i=1}^{N} (x^i, A_i x^i) - 2 \sum_{i=1}^{N} (c^i, x^i) \\ &+ 2 \sum_{i=1}^{N-1} b_i x_{1+3i} x_{3i} + 2z x_{3N} \Big]. \end{aligned}$$

We then have the following recurrence relation:

(6) $$\begin{aligned} f_N(z) = \operatorname*{Min}_{(x_{3N}, x_{3N-1}, x_{3N-2})} \Big[&(x^N, A_N x^N) + 2xz_{3N} \\ &- 2(c^N, x^N) + f_{N-1}(b_{N-1} x_{3N-2}) \Big]. \end{aligned}$$

35. Computational Aspects—I

Since the function $f_1(z)$ is readily determined, we can compute the sequence $\{f_k(z)\}$ at the expense of a minimization over a 3-dimensional region. This minimization may be greatly speeded up upon using the convexity properties of the functions involved. Although no optimal methods are known for multi-dimensional problems, the one-dimensional method presented in [36] may be employed in an iterative manner.

Writing (34.6) in the form

$$(1)\qquad f_N(z) = \min_{x_{3N-2}}\left[\min_{x_{3N},x_{3N-1}}\left[(x^N, A_N x^N) + 2zx_{3N} - 2(c^N, x^N)\right] + f_{N-1}(b_{N-1}x_{3N-2})\right],$$

we see that it reduces to

$$(2)\qquad f_N(z) = \min_y \left[g_N(z,y) + f_{N-1}(b_{N-1}y)\right],$$

where

$$(3)\qquad g_N(z,y) = \min_{x_{3N},x_{3N-1}}\left[(x^N, Ax^N) + 2zx_{3N} - 2(c^N, x^N)\right],$$

upon identifying x_{3N-2} as y. This new relation is now well-suited to the technique described in [3].

The computation of the functions $\{g_N(z,y)\}$ is independent of the computation of the sequence $\{f_N(z)\}$. Observe that this computational approach involves no divisions.

36. Computational Aspects—II

Another approach to the computational solution reposes upon the easily established fact that $f_N(z)$ is a quadratic in z for each N, i.e.

$$(1) \qquad f_N(z) = u_N + 2v_N z + w_N z^2,$$

where u_N, v_N and w_N are independent of z. This is the same device used in connection with Jacobi matrices, see [13].

Substituting in (34.6), we obtain the equation

$$(2) \qquad u_N + v_N z + w_N z^2 = \min_{(x_{3N}, x_{3N-1}, x_{3N-2})} \Big[(x^N, A_N x^N) + 2zx_{3N} - 2(c^N, x^N) + u_{N-1} + 2b_{N-1} x_{3N-2} v_{N-1} + b_{N-1}^2 x_{3N-2}^2 w_{N-1}\Big].$$

Upon performing the minimization and determining the minimum value of the right-hand side, we obtain recurrence relations connecting the triple (u_N, v_N, w_N) with the triple $(u_{N-1}, v_{N-1}, w_{N-1})$.

This affords an alternative computational technique.

The problem of determining the largest and smallest characteristic values may be approached in a similar fashion, cf. [13], [1].

37. Reliability of Multi-component Systems

A fundamental problem in the design of electronic systems, switching networks, computing devices and automata, is that of

constructing more reliable devices from less reliable components. Essentially, it becomes a question of minimal duplication. Some general discussions of these problems may be found in [17].

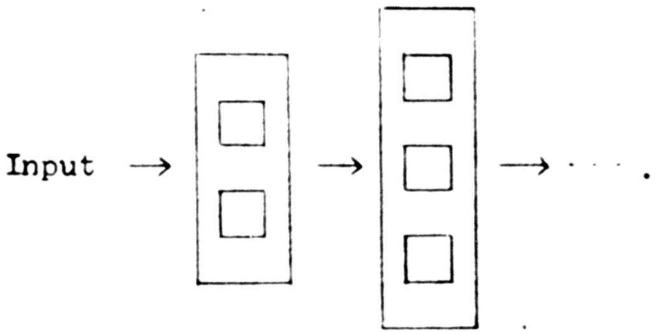
As a sample of the type of problem that can be treated using functional equation techniques, let us consider the following.

Let us suppose that the device we wish to design will consist of a number of stages each of which feeds its successor. Thus

$$\text{Input} \rightarrow \square \rightarrow \square \rightarrow \cdots \rightarrow \square \rightarrow \text{Output}.$$

The reliability of the device will be interpreted to be the probability that it operates successfully, and this in turn will be taken to be the product of the reliabilities of the individual stages.

In many cases, the overall reliability is too low for its intended use. One way to increase the reliability is to introduce a number of duplicate components in parallel at various stages. Thus



We assume that we possess switching techniques which will automatically introduce a new component into the circuit at any stage if the first component used is faulty. The reliability of the entire system can now be improved by duplication of components in this fashion.

The process cannot be carried to any logical extreme because of physical constraints of size and cost. Consequently, we have the problem of determining optimal duplication subject to given weight and size constraints. In addition to permitting first choice of the number of components at each stage, we shall subsequently also allow a choice of the type of component. This latter feature introduces combinatorial aspects, although, of course, these are already present in the condition of discreteness.

Assuming that at least one component must be used at each stage, let

(1) $p_j(x_j)$ = the probability of successful performance of the j-th stage if $1 + x_j$ components are used at the j-th stage.

Let the cost of a component at the j-th stage be c_j and the weight be w_j. Due to the increase in complexity of switching circuits as x_j is increased, there is no reason to assume proportional cost. However, we shall do so here to simplify the notation, since the method we present is equally applicable to the general problem.

The variational problem is then that of determining the maximum of

$$(2) \qquad \prod_{j=1}^{N} p_j(x_j),$$

subject to the constraints

$$(3) \quad (a) \quad \sum_{j=1}^{N} c_j x_j \leq c,$$

$$(b) \quad \sum_{j=1}^{N} w_j x_j \leq w,$$

$$(c) \quad x_j = 0,\ 1,\ \ldots\ .$$

It is clear that this problem may be transformed into that of computing a sequence of functions of two variables, as outlined in §9, and by use of the Lagrange multiplier technique reduced to a problem involving a sequence of functions of one variable. The details, and the results of some computations, may be found in [17].

38. Different Types of Components

Let us now suppose that at each stage we have a choice of

two types of components, those of an A-type and those of a B-type. Let $c_j(A)$, $w_j(A)$, be respectively the costs and weights for components of the A-type at the j-th stage, and $c_j(B)$, $w_j(B)$ denote the corresponding quantities for B-types.

Given the overall restrictions on weight and cost, we wish, as above, to determine the types of components, and the quantities, which maximize the total reliability. We shall suppose that the switching requirements are such that at any particular stage it is impossible to combine any number of A-components with any number of B-components.

Defining the sequence of functions, $f_1(c,w)$, in the usual fashion, with the functions $p_j(x_j,A)$, $p_j(x_j,B)$ corresponding to $p_j(x_j)$, we obtain the relations

$$\text{(1)} \qquad f_1(c,w) = \operatorname{Max}\begin{bmatrix} \operatorname*{Max}_{x_1} p_1(x_1,A) \\ \\ \operatorname*{Max}_{x_1} p_1(x_1,B) \end{bmatrix},$$

where $1 \leq x_1 \leq \operatorname{Min}\Big[[c/c_1(A)],\ [w/w_1(A)]\Big]$ in the expression containing A, and x_1 satisfies the analogous constraint in the expression containing B.

Generally,

$$\text{(2)} \qquad f_N(c,w) = \operatorname{Max}\begin{bmatrix} \left[\operatorname*{Max}_{x_N} p_N(x_N,A)f_{N-1}(c-c_N(A)x_N,w-w_N(A)x_N)\right] \\ \\ \left[\operatorname*{Max}_{x_N} p_N(x_N,B)f_{N-1}(c-c_N(B)x_N,w-w_N(B)x_N)\right] \end{bmatrix},$$

where x_N satisfies corresponding constraints,

(3) $$1 \leq x_N \leq \mathrm{Min}\Big[[c/c_N(A)],\ [w/w_N(A)]\Big],$$

$$1 \leq x_N \leq \mathrm{Min}\Big[[c/c_N(B)],\ [w/w_N(B)]\Big],$$

in the two maximizations.

39. Sequential Search

In the remainder of the paper we wish to discuss a number of interesting problems in which we encounter the general question of finding in minimum time an element of a finite set possessing certain distinguishing characteristics.

Considering the great importance of the problem and the fascinating nature of the questions that arise, it is amazing how little work has been done in the field.

40. Determining the Maximum Value of a Function

Let us begin with a simple deterministic problem. Given a continuous function $f(x)$ defined over the interval $[0,1]$, we wish to determine the value of x which maximizes $f(x)$.

For a variety of reasons, some of which we have discussed above, we do not wish to use calculus, but wish rather to employ a search method. To make the problem of determining the value of a maximizing x in an efficient fashion precise, let us pose the following problem.

"Given the continuous function $f(x)$ defined over $[0,1]$, determine the quantity $N(d)$ and the associated search policy so that one can guarantee that a maximum value can be located within an interval of length d in at most $N(d)$ steps."

If the function is taken merely to be continuous, with no additional properties, it is clear that $N(d) = 1/d$. If, however, we add that $f(x)$ is concave, then this number can be considerably reduced, and the problem possesses a very elegant solution.

This solution was found first by Kiefer, [41], and then, independently, and in a simpler fashion by Johnson, [36], using functional equations.

A similar problem can be posed with reference to locating the unique zero of a continuous, monotone, concave function. This has been resolved, using functional equations, by Gross and Johnson, [33].

A detailed discussion of this type of problem will be found in Kiefer, [42].

41. Sequential Testing

The problem we have discussed in the preceding section is a particular case of the general problem of sequential testing. Let us discuss two particular problems which will illustrate the difficulties in this domain.

Suppose that we have a piece of equipment which has N different parts to be examined if there is loss of function. Given a priori probability distributions associated with the individual parts, and a set of testing devices which furnish various indications, how should we proceed so as to locate all sources of malfunction in a minimum time?

One version of this problem arises in connection with de-

If the function is taken merely to be continuous, with no additional properties, it is clear that $N(d) = 1/d$. If, however, we add that $f(x)$ is concave, then this number can be considerably reduced, and the problem possesses a very elegant solution.

This solution was found first by Kiefer, [41], and then, independently, and in a simpler fashion by Johnson, [36], using functional equations.

A similar problem can be posed with reference to locating the unique zero of a continuous, monotone, concave function. This has been resolved, using functional equations, by Gross and Johnson, [33].

A detailed discussion of this type of problem will be found in Kiefer, [42].

41. Sequential Testing

The problem we have discussed in the preceding section is a particular case of the general problem of sequential testing. Let us discuss two particular problems which will illustrate the difficulties in this domain.

Suppose that we have a piece of equipment which has N different parts to be examined if there is loss of function. Given a priori probability distributions associated with the individual parts, and a set of testing devices which furnish various indications, how should we proceed so as to locate all sources of malfunction in a minimum time?

One version of this problem arises in connection with de-

these problems, we encounter very much the same difficulty that we met in the traveling salesman problem. As we proceed in our testing, the information pattern becomes tremendously complicated, and it appears to be impossible to describe the state of the system in any simple way.

43. Design of Experiments

The problems presented in the foregoing sections are in turn special cases of what may be called the general problem of the design of experiments.

We have considered initially the relatively simple case in which the structure of the system is assumed known. If the structure is taken to be partially unknown, we first encounter situations in which we hypothesize a stochastic structure, and then the very much more difficult situations in which we have to determine the structure on the basis of observation as we proceed.

The information we possess determines the decisions we make, and the decisions we make determine the new information pattern. The problem of determining optimal policies in situations of this type is very much more difficult than any of the problems we have previously discussed.

Not only is the analysis much more intricate because of the stochastic structure of the process, but it is no longer easy to make precise what we mean by an optimal policy.

For a detailed discussion of matters of this nature, we refer to the papers by Robbins, [47], Bellman and Kalaba, [22], Bellman, [14], and Karlin, [39], Karlin and Johnson, [38].

BIBLIOGRAPHY

1. Bellman, R., *Dynamic Programming*, Princeton University Press, Princeton, New Jersey, 1957.

2. Bellman, R., "A Functional Equation Arising in Allocation Theory," *Society of Industrial and Applied Mathematics*, Vol. 3, 1955, pp. 129-132.

3. Bellman, R., "Dynamic Programming and Lagrange Multipliers," *Proc. Nat. Acad. Sci. USA*, Vol. 42, 1956, pp. 767-769.

4. Bellman, R., "Some New Techniques in the Dynamic Programming Solution of Variational Problems," *Quart. Applied Math.*, to appear.

5. Bellman, R., "On a Routing Problem," *Quart. Applied Math.*, to appear.

6. Bellman, R., "On the Application of the Theory of Dynamic Programming to the Study of Control Processes," *Proc. Symposium on Control Processes*, Polytechnic Institute of Brooklyn, April, 1956, pp. 199-213.

7. Bellman, R., "Mathematical Aspects of Scheduling Theory," *Society of Industrial and Applied Mathematics*, Vol. 4, 1956, pp. 168-205.

8. Bellman, R., "On the Theory of Dynamic Programming—A Warehousing Problem," *Management Science*, Vol. 2, 1956, pp. 272-276.

9. Bellman, R., "Dynamic Programming and the Smoothing Problem," *Management Science*, Vol. 3, 1956, pp. 111-113.

10. Bellman, R., "Terminal Control, Time Lags, and Dynamic Programming," *Proc. Nat. Acad. Sci. USA*, Vol. 43, 1957, pp. 927-930.

11. Bellman, R., "Bottleneck Problems, Functional Equations and Dynamic Programming," *Econometrica*, Vol. 22, 1954.

12. Bellman, R., "On the Computational Solution of Linear Programming Problems Involving Almost Block Diagonal Matrices," *Management Science*, Vol. 3, 1957, pp. 403-406.

13. Bellman, R., "On Some Applications of Dynamic Programming to Matrix Theory," *Illinois Journal of Mathematics*, Vol. 1, 1957, pp. 297-301.

14. Bellman, R., "A Problem in the Sequential Design of Experiments," *Sankhya*, Vol. 16, 1956, pp. 221-229.

15. Bellman, R., and S. Dreyfus, *On the Computational Solution of Dynamic Programming Processes—VIII: A Bottleneck Situation Involving Interdependent Industries*, The RAND Corporation, Paper P-1282, April 17, 1957.

16. Bellman, R., and S. Dreyfus, *Approximations and Dynamic Programming*, The RAND Corporation, Paper P-1176, September 13, 1957.

17. Bellman, R., and S. Dreyfus, "Dynamic Programming and the Reliability of Multi-component Devices," *Jour. Oper. Research Soc. Amer.*, to appear.

18. Bellman, R., and S. Dreyfus, *On the Computational Solution of Dynamic Programming Processes—X: The Flyaway Kit Problem*, The RAND Corporation, Research Memorandum RM-1889, April 5, 1957.

19. Bellman, R., and S. Dreyfus, *On the Computational Solution of Dynamic Programming Processes—V: A Smoothing Problem*, The RAND Corporation, Research Memorandum RM-1749, April 2, 1957.

20. Bellman, R., and S. Dreyfus, *On the Computational Solution of Dynamic Programming Processes—II: On a Cargo Loading Problem*, The RAND Corporation, Research Memorandum RM-1746, November 5, 1956.

21. Bellman, R., and R. Kalaba, "On the Role of Dynamic Programming in Statistical Communication Theory," *IRE Transactions of the Professional Group on Information Theory*, Vol. IT-3, 1957, pp. 197-203.

22. Bellman, R., and R. Kalaba, "Communication Processes Involving Learning and Random Duration," *IRE National Convention Record*, 1958, *Section on Information Theory*, to appear.

23. Bellman, R., and R. Kalaba, "On the Principle of Invariant Imbedding and Propagation through Inhomogeneous Media," *Proc. Nat. Acad. Sci. USA*, Vol. 42, 1956, pp. 629-632.

24. Boldyreff, A., "An Iterative Technique for Determining Rail Capacity," *Operations Research*, 1957.

25. Cartaino, H., and S. Dreyfus, "Application of Dynamic Programming to the Minimum Time-to-climb Problem," *Aeronautical Engineering Review*, 1957.

26. Cairns, S., Balance Scale Sorting, The RAND Corporation, Paper P-736, September 7, 1955.

27. Dantzig, G., Application of the Simplex Technique to a Transportation Problem, Activity Analysis of Production and Allocation, Wiley and Sons, New York, 1953.

28. Dantzig, G., D. R. Fulkerson, and S. Johnson, "Solution of a Large-scale Traveling Salesman Problem," Jour. Oper. Research Soc. Amer., November, 1954.

29. Dreyfus, S., "An Analytic Solution of the Warehousing Problem," Management Science, October, 1957.

30. Ford, L. R., Jr., and D. R. Fulkerson, "A Simple Algorithm for Finding Maximal Network Flows and an Application to the Hitchcock Problem," Can. Jour. Math., Vol. 9, 1957, pp. 210-218.

31. Fulkerson, D. R., A Network Flow Feasibility Theorem with Applications to Incidence Matrices and the Subgraph Problem, The RAND Corporation, Paper P-1278, February 12, 1958.

32. Gaddum, J. W., A. J. Hoffman, and D. Sokolowsky, "On the Solution to the Caterer Problem," Naval Research Logistics Quarterly, Vol. 1, 1954, pp. 154-165.

33. Gross, O., and S. Johnson, Sequential Minimax Search for a Zero of a Convex Function, The RAND Corporation, Paper P-935, September 11, 1956.

34. Harris, T. E., and F. S. Ross, Fundamentals of a Method for Evaluating Rail Net Capacities, unpublished.

35. Jacobs, W., "The Caterer Problem," Naval Research Logistics Quarterly, Vol. 1, 1954, pp. 154-165.

36. Johnson, S., Best Exploration for Maximum is Fibonaccian, The RAND Corporation, Paper P-856, May 4, 1956.

37. Johnson, S., "Optimal Two- and three-stage Production Schedules with Set-up Times Included," Naval Research Logistics Quarterly, March, 1954.

38. Johnson, S., and S. Karlin, "A Bayes Model in Sequential Design," Annals of Math. Stat., December, 1956.

39. Karlin, S., and R. Bradt, "On the Design and Comparison of Certain Dichotomous Experiments," Annals of Math. Stat., Vol. 27, 1956, pp. 390-409.

40. Karush, W., "On a Class of Minimum-cost Problems," Management Science, Vol. 4, 1958, pp. 136-155.

41. Kiefer, J., "Sequential Minimax Search for a Maximum," Proc. Amer. Math. Soc., Vol. 4, 1953, pp. 502-506.

42. Kiefer, J., Jour. Soc. Ind. Appl. Math., 1957.

43. Kuhn, H., and A. W. Tucker, "Nonlinear Programming," Proc. Second Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, 1951.

44. Lehman, R. S., On the Continuous Simplex Method, The RAND Corporation, Research Memorandum RM-1386, November 24, 1954.

45. Lehman, R. S., Studies in Bottleneck Problems in Production Processes, Part II, The RAND Corporation, Paper P-492, 1954.

46. Prager, W., On the Caterer Problem, LBM-13, Division of Applied Mathematics, Brown University, 1956.

47. Robbins, H., "Some Aspects of the Sequential Design of Experiments," Bull. Amer. Math. Soc., Vol. 58, 1952, pp. 527-536.

48. von Neumann, J., "A Certain Zero-sum Two-person Game Equivalent to the Optimal Assignment Problem," Contributions to the Theory of Games, Annals of Mathematics Studies No. 28, Princeton University Press, 1953.

49. Wald, A., Statistical Decision Functions, John Wiley and Sons, 1950.